
Directives of Communicability: Towards Software Development Teams

Adriana Lopes¹, Clarisse Sieckenius de Souza², Tayana Conte¹

¹ USES – Grupo de Usabilidade e Engenharia de Software
Universidade Federal do Amazonas (UFAM) Manaus, AM – Brazil

{adriana, tayana}@icomp.ufam.edu.br

² Semiotic Engineering Research Group

PUC-Rio, Rio de Janeiro, RJ – Brazil.

clarisse@inf.puc-rio.br



USES Technical Report
TR-USES-2021-0001
March, 2021

Institute of Computing (IComp)
Federal University of Amazonas (UFAM)
Manaus, Amazonas 69077-000

ABSTRACT

The software artifacts developed in the early stages of the development process describe the proposed solutions for the software. For this reason, these artifacts are commonly used to support communication among members of the development team. Miscommunication through software artifacts occurs because practitioners typically focus on their modeling, without reflecting on how other software development team members interpret them. In this context, we proposed the Directives of Communicability (DCs) to support practitioners analyzing characteristics that affect the artifact's content on communication via artifact. We conducted two studies, with practitioners, to evaluate our proposal before transfer it to the software development industry. In this technical report, we present the material used in two studies.

1. Introduction

Artifacts developed in the early stages of the software development process, such as the different diagrams of the Unified Modeling Language (UML) (Freire et al., 2018; OMG, 2015), assist practitioners in understanding the problem for which software was required. As proposed solutions for software development are in artifacts, these artifacts also support team communication (Petre, 2013).

Communication is considered an important factor in software development, since miscommunication in software teams causes low productivity and software failures (Käfer, 2017). Miscommunication via artifact occurs, for example, when consumers (who take the information they see in the models for the development of another artifact) have different interpretations from the ones intended by the producers (who conceive the modeling of the soft-ware). As much as consumers know the modeling notation, the way the modeling is expressed by their producer can affect these practitioners' mutual understanding.

In order to mitigate miscommunication via artifact, we proposed the Directives of Communicability (DCs), presented in Lopes et al. (2019a). The DCs can mitigate this miscommunication by providing reflections to producers on their communication with other development team members.

Practitioners can use our proposal mainly in the artifacts developed in the initial stages of the development process, such as UML diagrams and mockups. We conducted preliminary studies to evaluate our proposal to reduce miscommunication (Lopes et al., 2019a; Lopes et al., 2019b). However, we noticed that new studies are necessary to evaluate the DCs concerning practitioners' perceptions before transferring them to the industry.

We conducted an exploratory study in order to evaluate the practitioners' perceptions of the DCs before transfer them to the industry. This study had the participation of 15 practitioners to carry out the modeling of a UML use case (OMG, 2015) with the support of DCs. The results of this study showed that the UML use cases developed, with the support of DCs, had little risk of obtaining miscommunication. In addition, the participants' perceptions about the DCs showed that they are useful to support communication via artifact and the directives contribute to the quality of software. However, it is also important to evaluate how DCs can be used in artifacts used by software development teams, identifying their application feasibility.

For this reason, we carried out a study in a software development team. This team used UML use case and mockups during the software development. This could help us to understand the necessary

improvements in our proposal before transfer the DCs to the industry. We conducted this study in a software team with 14 practitioners. Such practitioners worked on a cooperation project between the University of Brasilia and the Brazilian Army. The results of this study showed the potential of DCs to indicate improvements in the artifact's content.

The following sections present the questionnaires used in the two studies. After that, we describe the DCs used for UML use cases and mockups, both used in our two studies.

2. Questionnaire used in the Study 1

In this section, we describe the post-study questionnaire answered by the participants in the first study.

Name: _____ Time of experience in the software industry: _____

1. Please answer the questions below considering your experience with Directives of Communicability

| | Totally Agree | Strongly Agree | Partially Agree | Neutral | Partially Disagree | Strongly Disagree | Totally Disagree |
|---|---------------|----------------|-----------------|---------|--------------------|-------------------|------------------|
| My interaction with the Directives of Communicability is clear and understandable. | | | | | | | |
| Interacting with the Directives of Communicability do not require a lot of my mental effort. | | | | | | | |
| I find the Directives of Communicability easy to use. | | | | | | | |
| I find it easy to get the Directives of Communicability to do what I want it to do. | | | | | | | |
| Using the Directives of Communicability improve my performance better for understanding aspects of the software. | | | | | | | |
| Using the Directives of Communicability in my job has improved my productivity, since I will not to correct information that is not understood by colleagues. | | | | | | | |
| Using the Directives of Communicability enhances my effectiveness on communication with the team based on the artifacts. | | | | | | | |
| I consider the Directives of Communicability useful for software design. | | | | | | | |
| Assuming I had enough time to design software, I intend to use the Directives of Communicability. | | | | | | | |
| Considering that if I could choose any tool, I predict that I would use the Directives of Communicability. | | | | | | | |

2. What is your perception about the Directives of Communicability?

3. Questionnaire used in the Study 2

3.1 Questionnaire answered by the producers

We describe the post-study questionnaire answered by the producers in the second study.

Name: _____ Time of experience in the software industry: _____

Please answer the questions below considering your experience on communication via artifacts.

1. What is your role in the development team for this software project?

2. What artifacts are used as a means of communication in the team?

3. What is your opinion about this artifact as a means of communication?

4. If you want to report more information about communication via artifact, please describe below?

3.2 Questionnaire answered by the consumers

We describe the post-study questionnaire answered by the consumers in the second study. The main questions were based on the DCs.

Name: _____ Time of experience in the software industry: _____

Please answer the questions below considering your experience on communication via artifacts.

1. What is your role in the development team for this software project?

2. What artifacts are used as a means of communication in the team?

3. During the development of the project, do you believe that at some point the information in this artifact inconsistent with what you knew about the software?

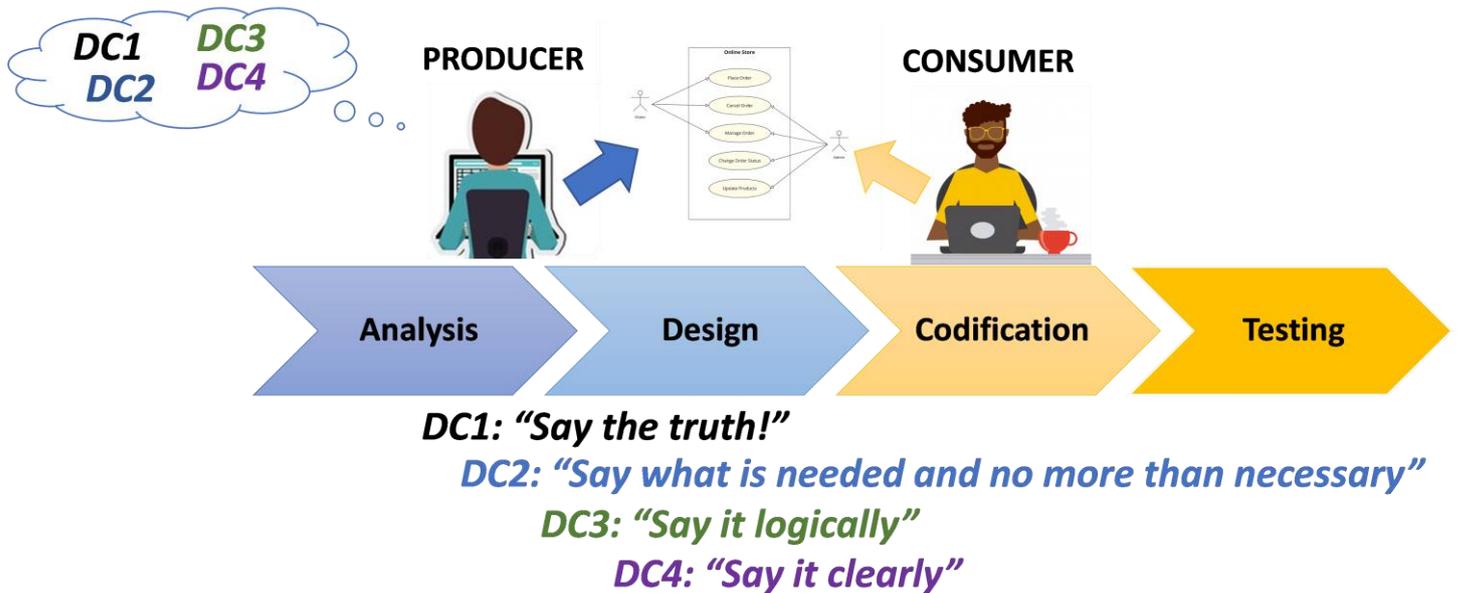
4. About the amount of information in this artifact for the understanding of the software, was there a lack of information? Or was there an excess of information?

5. Was there all the important information for the development of the software in this artifact?

6. Was there all the important information for the development of the software in this artifact?

7. Was there information more difficult to understand in this artifact?

DIRECTIVES OF COMMUNICABILITY



UML Use Case

DC1: "Say the truth!" - Use true information. Do not use information that affects the quality of the artifact.

- Are there uses cases that are not part of the problem domain?
- Do not insert outdated information

DC2: "Say what is needed and no more than necessary": Use the necessary content in the template. Do not use unnecessary content in the model

- Are all necessary use cases are there in the diagram?
- Insert all necessary relationships
- Do not use unnecessary content in the diagram
- Analyze the amount of information in the specification of all use cases

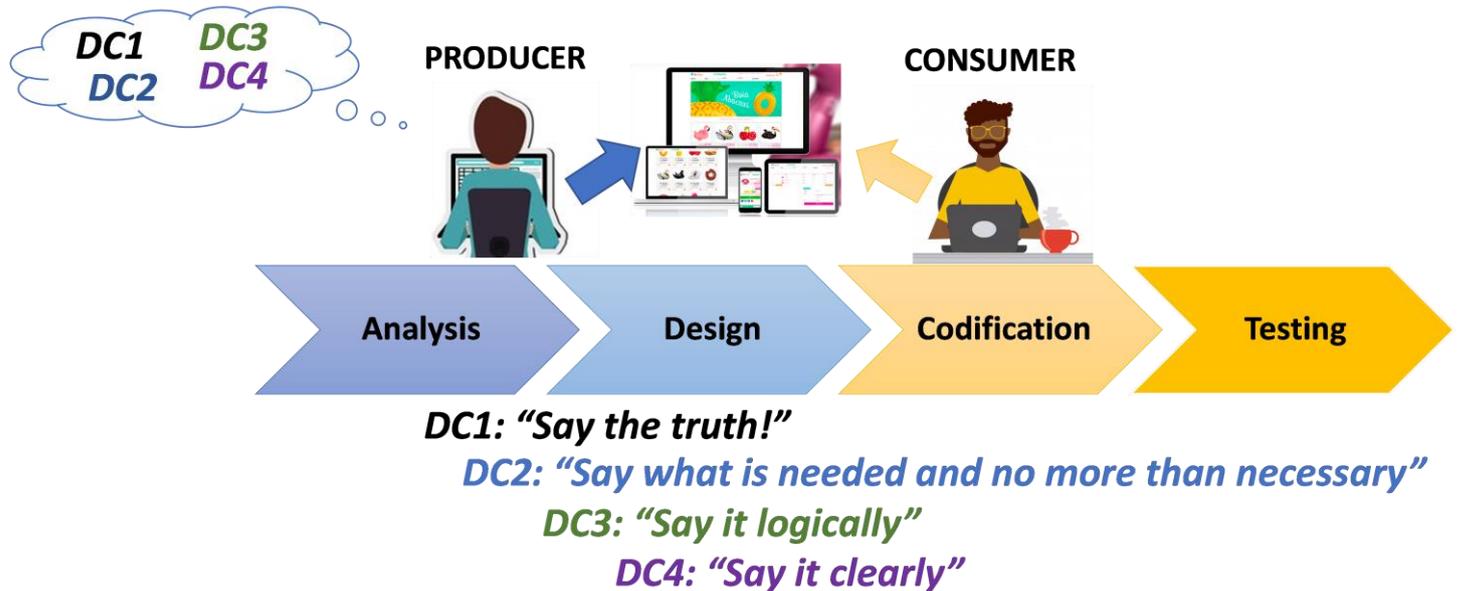
DC3: "Say it logically" - Organize the information in the model consistently

- Organize use cases in the diagram so that they are best viewed
- Organize relationships to represent a better view of the use cases in the diagram
- Organize the use cases in the specification so that they present a logical sequence for the consumers

DC4: "Say it clearly" - Organize the information in the model clearly

- Describe the names of the use cases so that they are easily understood and differentiated from each other
- Avoid ambiguity and implicit interpretation

DIRECTIVES OF COMMUNICABILITY



Prototypes

DC1: "Say the truth!" - Use true information. Do not use information that affects the quality of the artifact.

- Analyze if there is outdated information in the prototypes or that are outside the problem domain
- Do not insert outdated information

DC2: "Say what is needed and no more than necessary": Use the necessary content in the template. Do not use unnecessary content in the model

- Analyze whether the number of screens, with their respective interface elements, is sufficient for consumers to comprehend the system
- Do not use unnecessary content in the diagram

DC3: "Say it logically" - Organize the information in the model consistently

- Organize the screen in the specification so that they present a logical sequence for the consumers

DC4: "Say it clearly" - Organize the information in the model clearly

- Use terms in the content of the screens that can be easily comprehended and differentiated from each other
- Avoid ambiguity and implicit interpretation

References

- Freire, E. S. S., Oliveira, G. C., & de Sousa Gomes, M. E. (2018). Analysis of open-source CASE tools for supporting software modeling process with UML. In Proceedings of the 17th Brazilian Symposium on Software Quality, 51-60.
- OMG. (2015). Unified Modeling Language TM (UML) Version 2.5.
- Petre, M. (2013). UML in practice. In Proceedings of the 2013 International Conference on Software Engineering (ICSE 2013), 722-731.
- Käfer, V. (2017). Summarizing software engineering communication artifacts from different sources. In Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, 1038-1041.
- Lopes, A., Oliveira, E., Conte, T., & de Souza, C. S. (2019a). Directives of communicability: towards better communication through software models. In 2019 IEEE/ACM 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), 45-48.
- Lopes, A., Conte, T., & de Souza, C. S. (2019b). Reducing the risks of communication failures through software models. In Proceedings of the 18th Brazilian Symposium on Human Factors in Computing Systems, 1-10.